IN THE CLAIMS:

1. An object oriented computing system on a computer platform, comprising:

objects with semanticless, dynamically likable inputs and outputs; and

an event communication framework providing automated, pattern-based, fully distributable events.

2. The object oriented computing system of claim 1, wherein the inputs and outputs of the objects are provided via CsaConnectable and CsaRemote objects, respectively.

3. The object oriented computing system of claim 2, wherein each data structure associated with the inputs and outputs is described in a separate header file which can be used by every object to be linked.

4. The object oriented computing system of claim 2, wherein each object is a shared library which is dynamically likable at runtime by an ASCII configuration filing names of the inputs and outputs of the objects.

5. An object oriented computing system on a computing system, comprising :

objects having dynamically likable inputs and outputs and internal tasks for queuing of data transferred into and out from the objects via said inputs and outputs, respectively; and

an event communication framework providing automated, pattern-based, fully distributable events.

6. The object oriented computing system of claim 5, wherein the inputs and outputs of the objects are provided via CsaConnectable and CsaRemote objects, respectively.

7. The object oriented computing system of claim 6, wherein each data structure associated with the inputs and outputs is described in a separate header file which can be used by every object to be linked.

8. The object oriented computing system of claim 6, wherein each object is a shared library which is dynamically linkable at runtime by an ASCII configuration file containing names of the inputs and outputs of the objects.

9. A method for designing software components in an object oriented computing system, comprising the steps of:

defining input and output events that are fully distributable;

configuring dynamic linkable, semantic-free software components by input and output connections points; and

providing autorouted pattern based fully distributable events based on an event communication framework.

10. A storage medium including object oriented code having an object oriented computing system on a computer platform, comprising:

objects with semanticless, dynamically linkable inputs and outputs; and

an event communication framework providing automated, pattern-based, fully distributable events.

11. The storage medium of claim 10, wherein the inputs and outputs of the objects are provided via CsaConnectable and CsaRemote objects, respectively.

12. The storage medium of claim 11, wherein each data structure associated with the inputs and outputs is described in a separate header file which can be used by every object to be linked.

13. The storage medium of claim 11, wherein each object is a shared library which is dynamically likable at runtime by an ASCII configuration filing names of the inputs and outputs of the objects.

14. A storage medium, comprising:

object oriented code for an object oriented computing system on a computing system;

objects having dynamically linkable inputs and outputs and internal tasks for queuing of data transferred into and out from the objects via said inputs and outputs, respectively; and

an event communication framework providing automated, pattern-based, fully distributable events.

15. The storage medium of claim 14, wherein the inputs and outputs of the objects are provided via CsaConnectable and CsaRemote objects, respectively.

16. The storage medium of claim 15, wherein each data structure associated with the inputs and outputs is described in a separate header file which can be used by every object to be linked.

17. The storage medium of claim 15, wherein each object is a shared library which is dynamically linkable at runtime by an ASCII configuration file containing names of the inputs and outputs of the objects.

18. A method for designing software components in an object oriented computing system having a storage medium including object oriented code, comprising the steps of:

defining input and output events that are fully distributable;

configuring dynamic linkable, semantic-free software components by input and output connections points; and

providing autorouted pattern based fully distributable events based on an event communication framework.